AFRL-IF-RS-TR-2001-6
Final Technical Report
January 2001

REAL TIME CONTINUOUS SPEECH RECOGNITION FOR C3I APPLICATIONS

BBN Technologies

Marie Meteer, Christopher Barclay, Sean Colbath

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK**

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2001-6 has been reviewed and is approved for publication.

APPROVED: Richard M Evans

RICHARD M. EVANS
Project Engineer

FOR THE DIRECTOR:

ROBERT E. MARMELSTEIN, Maj, Deputy Chief
Information Systems Division
Information Directorate

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
JANUARY 2001

3. REPORT TYPE AND DATES COVERED
Final Sep 94 - Mar 98

4. TITLE AND SUBTITLE
REAL TIME CONTINUOUS SPEECH RECOGNITION FOR C3I APPLICATIONS

5. FUNDING NUMBERS
C - F30602-94-C-0086
PE - 62702F
PR - 5581
TA - 32
WU - 09

6. AUTHOR(S)
Marie Meteer, Christopher Barclay, Sean Colbath

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
BBN Technologies
70 Faucett Street
Cambridge MA 02138

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFSB
525 Brooks Road
Rome NY 13441-4505

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2001-6

11. SUPPLEMENTARY NOTES
Air Force Research Laboratory Project Engineer: Richard M. Evans/IFSB/(315) 330-2112

12a. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report describes the use of BBN's speech technology to provide support of ongoing Information Directorate programs in several areas, including research in intelligent interfaces and virtual reality. There were two major task areas: 1) to provide Hark Speech Recognizer product licenses to AFRL and their supporting contractors, and support them in the development of applications, and 2) develop tools and techniques to enhance the capability of the Hark Recognizer, focusing on collaboration and developing collaborative applications with voice input. The report also includes a collection of papers and presentation that have originated from this work.

14. SUBJECT TERMS
Speech Recognition, Collaborative Applications

15. NUMBER OF PAGES
28

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

List of Tables

1. Introduction

The goal of the project Real Time Continuous Speech Recognition for C3I Applications was to use BBN's speech technology to provide support of ongoing Rome Laboratory programs in several areas, including research in intelligent interfaces and virtual reality. There were two major task areas:

a) Provide Hark™ Speech Recognizer product licenses as directed by the COTR to Rome Laboratory and their contractors and support them in the development of applications.

b) Develop tools and techniques to enhance the capability of the Hark Recognizer.

Toward the first goal, licenses were given to the following sites and training was conducted on site at Rome Laboratory:

Richard Evans, RL/C3AB, Rome Laboratory, Griffiss AFB, NY
Steve Warzala, Syracuse University, Syracuse, NY
Sandy Pentland, MIT Medial Lab, MA
Mr. James Fleming, AL/HRTI, Brooks AFB, TX
Dr. Sam Schiflett, AL/CFTO, Brooks AFB, TX
Mr. Jason Moore, SUNY Binghamton, Binghamton, NY
Timothy R. Anderson, AL/CFBA Wright-Patterson AFB, OH

The remainder of the report describes the efforts in the second task area, in which we focused on collaboration and developing collaborative applications with voice input. We have also appended papers and presentations that have originated from this work.

2. Collaboration

Collaborative decision making is a central part of command and control, logistics, planning, and many other applications where multiple people work together to solve a common problem and need to be able to view the same information. There are two main ways of sharing visual information in a computer support environment. One is to have each person sitting in front of their own CTR viewing the same windows. The other is to have a central large screen that a number of people in a room can view together. While the former is the only viable alternative when participants are in remote sites, individual terminals can significantly cut down on the bandwidth of communication when people are in the same room. Eye contact and hand gestures cannot be used, and neither can the subtle protocols people have developed for turn taking in conversation. While this doesn't make communication impossible--we manage to communicate in groups in conference calls--it is far less efficient than face to face conversation. At Rome Laboratory such a large screen has been developed and at other sites applications that could take advantage of such a screen are being developed. MITRE is working on virtual reality in a command and control environment and battle simulation and TASC is working on collaborative environments for planning for both groups in a central location and at remote sites. While working in a group in front of a large screen clearly facilitates communication, there are many technical issues that need to be addressed before all of the features available in a computer support environment at a workstation are available for a large screen. One such technology is voice input. When sitting at a workstation, the microphone can be built in or attached to a headset by a cord. A push-to-talk button controlled by the mouse can be built into the interface and used easily in conjunction with other interface buttons. In a large screen environment, people are free to move around, which makes a stationary microphone or cord more cumbersome and a central button inconvenient. The goal of a computer support system is to integrate seamlessly into the environment, which in this case would be making the screen just another participant that someone could turn to and address just as he would someone else in the room.

The goal of this effort was to address some of the issues in developing voice input for a collaborative environment experimentally by building a sample application and testing the various parameters. Our targeted collaborative application is a disaster relief scenario running on a large shared screen. A number of technical issues surrounding collaboration in this environment have been explored. We first describe the application scenario, and then describe the different dimensions that we explored in the project, including the type of microphone, microphone mode (e.g., push to talk vs. continuous open mic), discourse models, dialog management, and using multiple applications.

2.1 The application scenario

BBN's OpenMap(tm) Map System is software that supports planning using maps and other types of geographical information. It provides a way to integrate different geographical data so that information can be combined and used in the planning process. It is both a mapping tool for the user and a system which provides mapping support for other software components. Collaboration is built into the base application. Accepting a conference invitation causes you to get a copy of all the maps in the conference and adds the maps you are working on into the conference as well. Maps in a conference are kept synchronized by the OpenMap server so that all participants see the same views. Once the conference is established, everyone sees the same information. There is a telepointer which can be used to point out interesting features while you are in a conference. Each user's telepointer is a different color. In addition to its collaborative facilities among groups of maps, OpenMap allows collaboration on a single map through the concept of layers, which allows users to create, in effect, mylar sheets which may draw and hide graphical information on the map. We are using data from the State of Hawaii denoting various governmental agencies devoted to disaster planning and crisis management. Disaster relief is an ideal collaborative application due to the fact that many governmental bodies (e.g., police, fire, rescue) must all work together towards a coordinated response. We wanted to explore how a speech interface would help users plan their scenario more easily and quickly.

2.2 Microphone evaluation

One of the most critical and controllable pieces of external equipment for speech recognition is the microphone. In a collaborative setting, one can envision using a centrally located microphone, or a microphone located somewhere on a user's body. We have evaluated the accuracy of different microphones against a baseline using the Sennheiser full headset microphone. An individual speaks a test phrase into both microphones simultaneously and a recognition result is generated for each microphone. We have discovered that the centrally mounted microphone (located off the speaker's body) has significant performance degradation (35-75% worse than the baseline) due to room noise and other factors. If mobility is an issue, we recommend using wireless lapel or headset microphones which do not suffer the same problems but which afford freedom of motion.

Microphone	Speaker	Control	Correct	Change
Telex Microphone	Tester1	80.10%	78.10%	2.00%
	Tester2	84.90%	83.60%	1.30%
	Tester3	81.90%	82.60%	-0.70%
	Tester4	84.00%	82.90%	1.10%
Shure Wireless Microphone	Tester1	86.00%	81.90%	4.10%
	Tester2	84.90%	80.90%	4.00%
Radio Shack Microphone	Tester1	83.30%	78.70%	4.60%
	Tester2	83.60%	87.40%	-3.80%
Jabra Earphone	Tester1	83.90%	57.10%	26.80%

Table 1 Results of Microphone Tests

In noisy environments, the Telex noise canceling microphone works much better than the baseline Sennheiser by filtering sounds not spoken within 1/4 inch of the microphone.

2.3 Microphone modes

We tested the following microphone modes:

2.3.1 Push to talk

The traditional push to talk button is implemented in this system. However, a user may wish to have a push to talk mode but not desire to push a button on the screen. We added the option of depressing the middle mouse button on the laser pointer to signal that the user would like to begin speech recognition. On the electronic whiteboard, this allows the user to be standing back at a distance (up to 5 feet).

2.3.2 Continuous speech

In continuous speech mode, the Recognizer is always listening and trying to interpret what is being said. This mode is appropriate in a quiet environment where a single user seated at a workstation is interacting with the system; however, it is inappropriate for collaborative environments where people are talking with each other in addition to talking to the computer.

2.3.3 Continuous speech with confirmation

A confirmation mode was added for use in noisy environments. Occasionally Hark misrecognitions will be unacceptably high. In these cases, a CONFIRM button provides the user to accept the recognized speech before an action is taken. This allows the user the opportunity to reject a misrecognition before a lengthy query is executed.

2.3.4 Continuous speech with keyword 'Computer'

One of the problems with the demonstration's continuous listening mode was synchronizing the user's start of speech with when Hark was listening. If the timing was off, occasionally speech would be rejected or misrecognized. The grammar was modified to allow an arbitrary amount of out-of-grammar speech to be spoken, followed by a pause and the keyword "computer", followed by in-grammar speech. This removed the synchronization problem with no apparent penalty in recognition accuracy. Users may speak simultaneously using the Dialog Manager referenced in Section 2.5.

2.4 Discourse Model

A discourse model is maintained of items and classes of items that are being acted upon by the participants in the collaboration. This model allows certain classes of items to be changed by using diectic reference, such as "those" or a descriptive phrase such as "north of Honolulu.

a) Color those rings green.
b) Delete the hospitals north of Honolulu.

The application provides shallow understanding of map manipulation commands (e.g., Zoom in) and deeper understanding database queries, (e.g., "Show me the hospitals within ten miles of Honolulu"). The discourse model is application specific and does not track individual user requests. For example, if User1 adds a group of red range rings, User2 may remove those rings. Further experimentation is needed on individual versus application discourse models. We would like to determine whether information a user has acquired from one application is helpful to other applications (e.g., location, quantities of items and type, etc.).

2.5 Dialog Management

When multiple people wish to interact simultaneously, the individual dialogs must be synchronized. We experimented with a single audio line into the system using a mixer, alternating speech among the participants. This worked well in small demonstrations but the potential in a collaborative environment for multiple simultaneous conversations exists. Additional sound cards may be acquired for most computers; we purchased one for the Sun workstation. A central application called the Floor Policy Manager (FPM) creates a speech recognition process for each user, where the amount of users is limited by the number of audio inputs and microphones. The FPM receives speech input or typed queries through a graphical display and sends the input to the application the user has selected. A current limitation of the system is that it can only process input from one mouse - this is a hardware limitation of most systems. Users register with the FPM and are assigned a microphone. Users may select applications by mouse (touching a grid of radio buttons) or by speech ("Go to application X"). Multiple users may interact with a single application or multiple applications.

2.6 Multiple Application Collaboration

A second application was added to retrieve weather and tsunami information off World Wide Web sites. This sets the stage for tests involving multiple users and multiple applications. This experimentation was not able to be completed in the time frame of this project.

2.7 Machine specifications

All applications have a socket interface, so they may be run on any available machine. Hark can run on a fairly small machine; we used a Sparc 5.

3. Direction of Future Work

On this project, we were able to set up the framework for exploring the issues in voice input for collaborative applications and begin the experimentation for how people can most effectively use this medium. The following is a set of tasks for future work that is designed to both extend the experimentation and transfer the technology to a broader set of applications.

1. Extend the existing demonstration system at BBN, which incorporates BBN's Hark and VISTA (Voice Input System To Applications) with the OpenMap™ system in a disaster relief scenario.

2. Develop technology to allow for multiple channels and multiple recognizers so more than one person can talk at once.

3. Develop a dialog module to allow different people to refer to the same things using pronouns (e.g., P1: "Show the hospitals" P2: "Highlight them in red", where "them" refers to the hospitals).

4. Develop techniques for integrating multiple pointing devices, keeping track of which speakers are using which pointer, and integrating language and pointing (e.g., "Highlight this one in red").

5. Integrate techniques for providing feedback to the user, both visual and verbal (e.g., using speech synthesis or recorded speech) and evaluate which are most effective.

6. Develop and evaluate protocols for switching applications and contexts, i.e., individuals moving from one "conversation" to another, or starting a new conversation with another participant. Also determine the appropriate behavior of "cancel" and "undo" commands in a collaborative environment.

7. Continue analysis of microphones, system configurations, and other parameters particular to collaborative applications.

8. Deliver the disaster relief demonstration system to Rome Laboratory to be shown on the RL demonstration data wall.

9. Support the integration of speech and language understanding into JFACC applications on the Sparc based data wall, including developing speech and language grammars for the domains being used and providing tools for extending those grammars and dictionaries.

10. Ensure that all of the technology developed under this effort will be able to be transferred to the portable data wall environment on the PC.

11. Deliver Hark and Hark development tools to Rome Laboratory contractors.

Appendix A

Collaboration Using Spoken Language and Traditional Interfaces

Christopher Barclay and Sean Colbath
cbarclay@bbn.com, scolbath@bbn.com
BBN Technologies
70 Fawcett St.
Cambridge, MA 02138
(617) 873-4815

0. Abstract

Collaboration requires the dissemination of knowledge among multiple participants tackling a common problem. This process is generally iterative as more information is brought to bear on the problem and issues are identified and resolved. Collaboration often takes place in front of a shared writing surface such as a white board; important points and ideas are noted here. As more information is obtained from computer databases and other non-traditional repositories of information (e.g. the World Wide Web), the computer has been incorporated into the collaborative process as an aide to the participants. An alternative idea is to establish the computer as another participant in the conversation, albeit one that will only "speak" when spoken to and will provide supporting facts. The computer will assist in providing and visualizing information.

1. Project Goals

For the computer to have a role in the conversation, it needs to have an intuitive way for people to communicate with it. BBN Technologies has developed an electronic whiteboard which functions both as a computer screen and an input device. The electronic whiteboard uses a rear projection system to display images onto a surface that is touch sensitive. Electronic pointing devices emit a distinct signal for each mouse button, enabling all the functionality of the computer except keyboard input. While this is sufficient for some applications, we find retrieving information from databases is frequently problematic without a keyboard. Given a choice when standing in front of a whiteboard, users would prefer not to have to hold a keyboard. Spoken language brings the computer into the dialog as a full participant. The computer listens to the conversation and responds when a question is posed.

Our goal was to develop a system in which all collaborators could speak to other participants, to a shared collaborative application or to their personal application(s). The goal requires a robust speech recognition system that can discard information that is not directed at the computer, does not require training by any of the participants (i.e., speaker independent), and which people can speak to in a natural manner (i.e., continuous speech recognition). BBN's Hark speech recognition system meets these goals for medium vocabulary (3000 word) applications.

2. The Speech System

Just as humans have trouble understanding when more than one person is talking simultaneously, speech recognition works most accurately when conversations are synchronized. Restricting conversations to a serial one-at-a-time approach seemed to be a potentially onerous demand since there may be multiple conversations involving multiple participants, and in a truly collaborative environment standard turn-taking speech patterns may not be followed. In some cases, however, the state-of-the-art may require some limitations on spoken interaction with the system.

To solve this problem, the speech interface to the collaborative environment was designed with flexibility in mind. Participants have a choice of using head-mounted or hand-held microphones. The microphones may be wireless or tethered. Each microphone is connected to a separate sound card on the host computer, with a separate recognizer listening to it. This allows the collaboration system to "listen" to each user individually.

The collaboration system needs to know when it is being addressed, instead of one of the other users (or another person in the room). Several distinct microphone modes are allowed, giving the users of the system the ability to trade off convenience and naturalness of interaction with very high accuracy and reduced chance false recognition (a tradeoff that might be made if the environment is particularly noisy). The microphone modes include:

Push-to-talk

The traditional push to talk button is implemented in this system. However, a user may wish to have a push to talk mode but not desire to push a button on the screen. We added the option of depressing the middle mouse button on the remote pointer to signal that the user would like to begin speech recognition. On the electronic whiteboard, this allows collaboration by users who are not immediately in front of the whiteboard (anywhere from 5-10 feet away).

Continuous speech with keyword ("COMPUTER")

Frequently used in Hollywood science-fiction movies, this microphone mode can actually be used with reasonable reliability in a low-to-medium noise environment. We implemented this by creating a hand-tuned rejection grammar that required all commands to the collaboration system be prefixed by a half-second pause and the "Computer" keyword. A rejection grammar is a grammar that exists in parallel to the grammar of acceptable commands, except that it consists of "noise" utterance including utterances like those acceptable to the system. The only path into the "good" grammar is one prefixed by the silence/keyword pair, and the grammar is heavily weighted to go into the rejection half of the grammar (and likely stay there). This means that the recognition system must have a fairly high confidence of hearing the keyword before it will emit a recognized string.

Continuous speech without keywords

This is generally useful in a quiet setting with few participants. Everything uttered by the user is interpreted as a command, without any attempt to reject out-of-grammar speech. This does not allow speech between the human collaborators without temporarily taking the microphone out of this mode.

Speech with confirmation

A confirmation mode was added for use in noisy environments to allow the user to explicitly accept the recognized speech before an action is taken.

3. Language Understanding

Speech recognition provides text for what users say, but this is insufficient for extracting the meaning that most collaborative applications require. The speech recognition system must be backed up by a language understanding system which holds a discourse model for the application.

The discourse model is maintained of items and classes of items that are being acted upon by the participants in the collaboration. The model allows users to refer to items using simple pronouns such as "this" or "that" while pointing with the remote pointer or complex referring such as "the green ones", or "the one to the left of the shelter".

The discourse model is inherently application specific. It needs to be able to get access to the data used by an application so that it can make quantitative decisions about data ("show me the cities with more than five fire stations"), and so that it can interact with the other modalities provided by the application. For instance, statements such as "here" and "there" require the ability to extract any area selected by the mouse or pointing device.

We built our discourse model using the SIMS system, produced by the Information Sciences Institute (ISI), at the University of Southern California. SIMS allows easy construction of natural language understanding systems that tie into both applications and relational databases. This is done by constructing a source and a domain model. The source model is used to describe the sources of data (typically, a relational database such as Oracle) and attributes (typically, via communication with an application) in the discourse system, and the domain model is used to describe the ways that they can be expressed by the user – in natural human language. SIMS finds relationships between the two models and provides answers to the user (or application) by extracting data from databases according to these relationships.

4. The Application, Scenario, and Task

In order to demonstrate and evaluate the collaboration system, we needed to build an application that we could use, as well as describe a plausible scenario under which one or more users would be collaborating to solve a problem, and a task that they would be doing while working on the problem.

The scenario we came up with was that of a disaster relief operation on the island of Oahu after a natural disaster. Disaster relief requires the cooperation of many groups of people (military, civilian, and governmental) across a wide variety of functions (medical, fire, construction, etc.). Working with a map to visualize data is a common task performed by people working on similar problems.

We chose BBN's OpenMap™ System to build our first collaborative application. OpenMap™ is software that supports planning using maps and other types of geographical information. It provides a way to integrate different data with spatial attributes so that information can be combined and used in the planning process. It is both a mapping tool for the user and a system which provides mapping support for other software components. A more conventional form of networked collaboration is built into the base application. Accepting a conference invitation causes the user to get a copy of all the maps in the conference and adds the maps the user is working on into the conference as well. In addition to its collaborative facilities among groups of maps, OpenMap™ allows collaboration on a single map through the concept of layers, which allow users to create sheets on which they may draw and hide graphical information on the map.

Our application is a mockup of a logistics application which simulates a disaster relief effort. Participants must solve problems such as coordinating local relief, moving people to shelters, determining where sufficient hospital beds are and whether the hospitals have appropriate facilities to treat the injured.

In addition to a traditional mouse-pointer interface via the pointer for the whiteboard, we implemented several classes of speech commands:

Map-driven command-and-control

The most basic type of speech command; although these could be implemented as macros, they do pass through the entire understanding system. Examples of these commands would be “ZOOM IN”, or “GO TO OAHU”.

Multi-modal

These commands mix input from the speech system and the mouse. An example of a multi-modal command would be “GO HERE” after the user selected an area on the map via rubberbanding, or “WHAT IS THE SUPPLY LEVEL AT THAT LOCATION” after the user highlighted a hospital via clicking on it. The understanding system has access to the application state including that of its graphical user interface, so commands like this are possible.

Data-driven

The most sophisticated kind of speech command is used to access the data that is being displayed on the map. Commands were developed to allow a wide variety of quantitative and relational questions about the data, which consisted of population of cities, capacity of emergency shelters and hospitals, location of particular resources (fire trucks, toxic waste cleanup equipment), locations of buildings (police stations) and the like. Sample questions include “SHOW THE HOSPITALS WITH CAPACITY GREATER THAN ONE HUNDRED FIFTY”, or “DISPLAY THE ROADS EAST OF WAIKIKI”. Once objects have been placed on the screen, they can be referred to. An example command might be “DRAW A THREE KILOMETER RED RANGE RING AROUND THOSE HOSPITALS”, where “THOSE” referred to the hospitals in the first utterance.

5. Usage

During use of the system, the users don the wireless headsets and approach the whiteboard. The screen of the whiteboard typically displays the OpenMap™ application covering most of the screen, as well as a small control panel with a listen button, a small window for recognition and status output, an indication of what users are logged in to the system, what application they are “talking” to, and the currently chosen microphone mode.

The users may take turns talking to the system, as would be natural, or they may talk simultaneously to each other or to other human collaborators in the room without headsets. OpenMap™ provides the ability to create “map layers,” virtual versions of the clear sheets of plastic placed over maps in many situation rooms allowing users to “write” directly on the map. These map layers can be created, hidden, ordered, and destroyed via voice, allowing the collaborators to explore many different solutions without having to do extensive backtracking or destroying each other’s work.

Due to various hardware limitations, only one remote pointer can be used with the system, but since the whiteboard is simply a projection screen, another user may sit at the projected computer's console and participate from there, using the native mouse on the system.

6. Future Work

The collaboration system was built as a pilot project to demonstrate the use of spoken language in a novel collaborative environment. The system was built to allow us to continue with a wide variety of experiments, such as how the architecture scales with application complexity, vocabulary size, and number of users and noise in the room.

We would also like to experiment with the discourse system, allowing participants to maintain discourse among distinct applications. For instance, it would be natural to talk with an application that knew about weather in particular cities and ask it to provide you with a forecast, and then return to the mapping application and refer to the previously mentioned city as "THERE".

Another interesting approach that we would like to explore is to blend in more traditional forms of networked collaboration (such as that provided by OpenMap™ and other systems), allowing users to participate from remote sites, and allowing the remote users to use speech to access the application.

While we intend to provide a copy of the demonstration to the customer, it would be good to build a second-generation system using an actual customer application, instead of one contrived for experimental work.

7. Acknowledgments

This work was funded by the Air Force Rome Laboratory under contract F306502-94-C-0086.

The authors would also like to thank Cecile Pham, Marie Meteer, and Rusty Bobrow for their invaluable contributions to this effort.

8. References

Ambite, Jose-Luis, Arens, Yigal, et al. "The SIMS Manual Version 1.0," Information Sciences Institute, University of Southern California

Milazzo, Paul. "A Collaboration Facility for BBN". BBN Internal Technical Report.

APPENDIX B

Designing Conversational Interfaces for Mobile Computing

Josh Bers Scott Miller John Makhoul
BBN Systems & Technologies
70 Fawcett St. Cambridge, MA 02138

1. Introduction

Next generation mobile networked systems will use human conversational modalities to interact with the user. In order to emulate more natural ways of communicating, keyboards and mice need to be replaced by speech and gesture based interfaces. Speech enabled browsing offers an attractive alternative to the current hunt-and-click approach. It also provides a richer way of accessing the growing amount of on-line information. In this direction, the current research focuses on providing conversational interfaces to on-line applications through speech recognition technology. We have developed a prototype system that combines pen and speech input from the on-line user in a web-browser. VoiceLog is a voice-activated system for obtaining parts diagrams and placing orders for vehicle parts from a web-server. The system runs on a mobile keyboard-less computer with a microphone and a wireless connection to the Internet. VoiceLog uses a novel client-server approach to speech recognition suitable for dynamic web-based applications.

The use of speech recognition to increase efficiency of performing tasks has been a priority of computer-human interface research [1,2]. Recent studies have shown the benefits of integrating multiple modes of input, i.e., pen/mouse and speech in graphical applications[3]. The user of VoiceLog selects objects with the pen/mouse and specifies an action through speech; for example, the user says "show me detail," while pointing to an object on the display[4, 5].

2. System

VoiceLog demonstrates the advantages of speech and pen on a mobile networked device. No keyboard is necessary; commands and objects are referenced through either speech or pointing. An example interaction follows:

U: Show me the hum-vee.
S: (displays image of the HMMWV)
U: Detail the engine.
S: (The area of the engine flashes and a part diagram of the engine is displayed)

U: Expand this area (user points at a region with the pen).
S: (area is highlighted and an expanded view of the part (fuel pump) is shown)
U: I need 10 (points to a specific screw).
S: (brings up an order form with the part name, part number and quantity fields filled in)

3. Interface

We chose a Java applet running in a web browser as the platform because of its ubiquity and user familiarity. The VoiceLog web-page is divided into two frames. The small upper frame always contains the system status area for textual feedback as well as GUI buttons for control and status of the recognizer. The larger lower frame displays either images from the parts catalog or an order form. Catalog images contain "hot" regions corresponding to individual parts that may be selected with the pen or through speech. Objects flash on the image when selected. The order form allows multimodal filling by pointing to a slot and speaking a value or writing it in with the pen. Modified items subsequently appear in boldface on the order form.

The speech system allows two modes of interaction, click-to-speak and hands-free, continuous, mode where the user can give multiple sequential commands without touching the screen to reactivate recognition.

4. Architecture

The system's functional breakdown reveals three major components: speech input, pen input, and integration and response (application). The architecture of the system allows each input modality to run in its own thread. This permits a simple overall design and enables quick feedback to the user. For example, when the user selects a region with the pen, the area flashes, and when the user says something that the system didn't recognize it will respond orally, "excuse me?" This low-level feedback is handled by the input threads, leaving the application free to process and give feedback to application-level events.

For the speech input, we created a networked server application based on BBN's Hark 3.0 recognizer that allows concurrent clients with independent grammars. This server splits the computational workload of speech recognition up so that the client machine performs the front end signal processing and sends the vector-quantized signal to the server for decoding into words using HMM's [6]. This centralized speech-server approach permits low-end client machines, and facilitates grammar/vocabulary updates - both of which are crucial for dynamic, web-based applications.

The pen input thread enables the user to select regions from a graphic image. Feedback to the user includes displaying the name of the object pointed at and highlighting the selected region.

The application thread receives events from the two input threads and performs actions appropriate to the current state. There are four major states of interaction plus two repair states: Idle: waiting for input; Object_input: object selected, waiting for command selection; Action_input: received a command, waiting for object selection; and Both_input: object and command specified, ready to take action, waiting for additional refinement of command. The two repair states handle failures to supply either an object, or an action. Time-outs in each state maintain the dialogue flow. For example, once the system times-out in the Both_input state it prints the interpretation of the user's input in the status window, i.e., "ordering 6 wheels," and then it performs the appropriate action, e.g., displays the order form. If the user forgets to choose an item for an order command, the system will prompt orally, "order which object?" and display the incomplete command in the status window, "order 6 ___". Recognition is then activated so that the user can then select an object with either speech or pen input.

5. Future Work

Currently the VoiceLog system is running on UNIX workstations, desktop PC's and a mobile, pen-based PC. Our future plans for this system are to leverage other technologies: speaker verification for order authorization and text to speech generation for more conversational interaction. We would also like to perform user studies to evaluate our system and to compare with traditional interfaces. Other future work includes enhancing both the language element and the gesture input for more powerful interactions.

6. References

[1] Martin, G. L. The utility of speech input in user-computer interfaces. International Journal of Man-Machine Studies, 1989. 30: p.355-375.

[2] Bowman, Maj. T. An evaluation of the utility of a computer speech recognition interface such as the voice activated logistics anchor desk and its relationship to decision making techniques for army logisticians. Master's Thesis, The Command and General Staff College, Ft. Leavenworth, KS, 1996.

[3] Oviatt, S. L., A. DeAngeli, and K. Kuhn. Integration and synchronization of input modes during multimodal human-computer interaction. Proceedings of CHI'97 Human Factors in Computing Systems (March 22-27, Atlanta, GA), ACM Press, New York.

[4] Bolt, R. A., Put-that-there: voice and gesture at the graphics interface. Computer Graphics, 1980. 14(3): p.262-270.

[5] Koons, D., Sparrell, C. & Thorisson, K. Integrating simultaneous input from speech, gaze and hand gestures, Intelligent Multimedia Interfaces, ed. by M. Maybury, MIT Press: Cambridge, MA, 1993, 257-76.

[6] Stallard, D. SPIN: a general architecture for speech applications on the internet. (in submission)

MISSION
OF
AFRL/INFORMATION DIRECTORATE (IF)

The advancement and application of Information Systems Science and Technology to meet Air Force unique requirements for Information Dominance and its transition to aerospace systems to meet Air Force needs.